Implementing a Fourth Order Runge-Kutta Method for Orbit Simulation

C.J. Voesenek

June 14, 2008

1 Introduction

A gravity potential in spherical harmonics is an excellent approximation to an actual gravitational field. Using a computer programme, orbits in this gravity potential can be simulated. This gives a good assessment of orbit disturbances due to irregularities in the gravitational field in question.

A good numerical method is required to accurately and quickly perform an orbit simulation. Simple methods (e.g., Euler, modified Euler) need a very small time step, and thus a large amount of computing time, to remain stable. Therefore, a more elaborate method is necessary to increase accuracy and to reduce calculation time. A fourth order Runge-Kutta method (RK4) is very well suited for this purpose, as it is stable at large time steps, accurate and relatively fast.

2 Fourth order Runge-Kutta method

The fourth order Runge-Kutta method can be used to numerically solve differential equations. It is defined for any initial value problem of the following type.

$$\begin{aligned} y' &= f(t, y) \\ y(t_0) &= y_0 \end{aligned} \tag{1}$$

The definition of the RK4 method for the initial value problem in equation (1) is shown in equation (2).

$$y_{n+1} = y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) \tag{2}$$

With h the time step, and the coefficients k_1, k_2, k_3 and k_4 defined as follows.

$$\begin{aligned} k_1 &= f(t_n, y_n) \\ k_2 &= f(t_n + \tfrac{h}{2}, y_n + k_1\tfrac{h}{2}) \\ k_2 &= f(t_n + \tfrac{h}{2}, y_n + k_2\tfrac{h}{2}) \\ k_2 &= f(t_n + \tfrac{h}{2}, y_n + k_3 h) \end{aligned} \tag{3}$$

These coefficients indicate the slope of the function at three points in the time interval: the beginning, the mid-point and the end. The slope at the midpoint is estimated twice, first using the value of k_1 to determine k_2, next using the value of k_2 to compute k_3.

Knowing the k-coefficients, the solution at the next time step can be computed with equation (2).

3 Application to orbits in a gravity potential

Since the problem of orbit simulation involves a three-dimensional, second order differential equation, the method will have to be slightly modified to be useful.

The gravitational potential U of the object can be expressed in terms of the coordinates x, y and z. With this, equations (4), (5) and (6) can be used to compute the accelerations in all directions for the current position.

$$\ddot{x} = -\frac{\partial U}{\partial x} \tag{4}$$

$$\ddot{y} = -\frac{\partial U}{\partial y} \tag{5}$$

$$\ddot{z} = -\frac{\partial U}{\partial z} \tag{6}$$

The position vector $\vec{r}$, velocity vector $\vec{v}$ and acceleration vector $\vec{a}$ of the object in the gravity potential can be defined as follows.

$$\vec{r} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} \quad \vec{v} = \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} \quad \vec{a} = \begin{pmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{pmatrix}$$

The second order differential equation governing the motion of the orbiting object can be expanded into the two first order differential equations (7) and (8).

$$\dot{\vec{r}} = \vec{v} \tag{7}$$

$$\dot{\vec{v}} = \vec{a} \tag{8}$$

For these equations, the k-coefficients can be defined. These coefficients are given in vector form as the slopes are estimated for each coordinate separately.

Their definitions for differential equation 7 are given in equation 9.

$$\begin{aligned} \vec{k}_{1_{v_{i+1}}} &= \vec{a}\left(\vec{r}_i\right) \\ \vec{k}_{2_{v_{i+1}}} &= \vec{a}\left(\vec{r}_i + \vec{k}_{1_{r_{i+1}}} \frac{h}{2}\right) \\ \vec{k}_{3_{v_{i+1}}} &= \vec{a}\left(\vec{r}_i + \vec{k}_{2_{r_{i+1}}} \frac{h}{2}\right) \\ \vec{k}_{4_{v_{i+1}}} &= \vec{a}\left(\vec{r}_i + \vec{k}_{3_{r_{i+1}}} h\right) \end{aligned} \tag{9}$$

The acceleration vector $\vec{a}$ can be computed using equations (4), (5) and (6) for each of the slopes.

The k-coefficients for the second differential equation (8) are defined as follows.

$$
\begin{aligned}
\vec{k}_{1_{r_{i+1}}} &= \vec{v}_i \\
\vec{k}_{2_{r_{i+1}}} &= \vec{v}_i \vec{k}_{1_{v_{i+1}}} \frac{h}{2} \\
\vec{k}_{3_{r_{i+1}}} &= \vec{v}_i \vec{k}_{2_{v_{i+1}}} \frac{h}{2} \\
\vec{k}_{4_{r_{i+1}}} &= \vec{v}_i \vec{k}_{3_{v_{i+1}}} h
\end{aligned}
\tag{10}
$$

The coefficients need to be computed alternately for the velocity and the position, since the first k-coefficient is required to compute the second, et cetera.

When all intermediate slopes are calculated, the velocity and position vector at the next time step can be determined using respectively equation (11) and (12).

$$\vec{v}_{i+1} = \vec{v}_i + \frac{h}{6}\left(\vec{k}_{1_{v_{i+1}}} + 2\vec{k}_{2_{v_{i+1}}} + 2\vec{k}_{3_{v_{i+1}}} + \vec{k}_{4_{v_{i+1}}}\right) \tag{11}$$

$$\vec{r}_{i+1} = \vec{r}_i + \frac{h}{6}\left(\vec{k}_{1_{r_{i+1}}} + 2\vec{k}_{2_{r_{i+1}}} + 2\vec{k}_{3_{r_{i+1}}} + \vec{k}_{4_{r_{i+1}}}\right) \tag{12}$$

The obtained velocity and position can be used to perform the same computation for the next time step. Thus, it is possible to obtain a complete solution over a certain time domain.

The described method can be easily implemented in any programming language. The resulting programme allows for quick and reliable calculations of orbits in any gravity potential.

$$- \frac{(1-\mu) y_1}{r_1^3} - \frac{\mu y_1}{r_2^3} + y_1 - 2 v_x$$

$$- \frac{(1-\mu)(x+\mu)}{r_1^3} - \frac{(\mu) \cdot (x - (1-\mu))}{r_2^3}$$

$$+ x + 2 v_y$$

$\vec{a} = F(\vec{x}, \vec{v})$

$\vec{V} = \vec{a}\, dT$ (1st order)

$\vec{X} = \vec{v}\, dT$ (1st order)

→ Euler Not Energy Conserving

$a_x = F(v_y, x, \mu, r_1, r_2)$

$a_y = F(v_x, y, \mu, r_1, r_2)$

Better

$x_0, y_0, v_{x0}, v_{y0} = (\vec{X}_0, \vec{V}_0)$

$k\vec{X1} = \vec{V}_0 \Delta T$

$k\vec{V1} = a(\vec{X}_0, \vec{V}_0)\Delta T$

$k\vec{X2} = \left(\vec{V}_0 + \frac{a(\vec{X}_0, \vec{V}_0)\Delta T}{2}\right)\Delta T = \left(\vec{V}_0 + \frac{k\vec{V1}}{2}\right)\Delta T$

$k\vec{V2} = \left(a\left(\vec{X}_0 + \frac{\vec{V}_0 \Delta T}{2}, \vec{V}_0 + \frac{a(\vec{X}_0, \vec{V}_0)\Delta T}{2}\right)\right)\Delta T = a\left(\vec{X}_0 + \frac{k\vec{X1}}{2}, \vec{V}_0 + \frac{k\vec{V1}}{2}\right)\Delta T$

$k\vec{X}3 = \left(\vec{V}_0 + \frac{k\vec{V2}}{2}\right)\Delta T$

$k\vec{V3} = a\left(\vec{X}_0 + \frac{k\vec{X2}}{2}, \vec{V}_0 + \frac{k\vec{V2}}{2}\right)\Delta T$

$k\vec{X4} = \left(\vec{V}_0 + k\vec{V3}\right)\Delta T$

$k\vec{V4} = a\left(\vec{X}_0 + k\vec{X3}, \vec{V}_0 + k\vec{V3}\right)\Delta T$

$$\vec{X}_n = \vec{X}_0 + \frac{1}{6} kx1 + \frac{1}{3} kx_2 + \frac{1}{3} kx3 + \frac{1}{6} k\vec{x}4$$

$$\vec{V}_n = \vec{V}_0 + \frac{1}{6} kv1 + \frac{1}{3} k\ v2 + \frac{1}{3} kv3 + \frac{1}{6} kv4$$

$|r_1| + |r_2| = 1,\ |r_2 - r_1| = 1,\ \ m_1 + m_2 = 1,$

$\vec{r}_1 = 1-\mu$	$m_1 = \mu$
$\vec{r}_2 = -\mu$	$m_2 = 1-\mu$

$\mu = \frac{m_1}{m_1 + m_2} = m_1$

Ex $\mu = .2$ $m_1 = .2$ $r_1 = .8$

$m_2 = .8$ $r_2 = -.2$

m_2 m_1

$r_1^2 = (x - 1 + \mu)^2 + y^2$

$r_2^2 = (x + \mu)^2 + y^2$

Now $G = 1,\ w = 1$ $\ddot{\vec{r}} - w\vec{r} - 2w\vec{v} = \frac{-\mu}{r_1^3}(\vec{r} - \vec{r}_1) - \frac{(1-\mu)}{r_2^3}(\vec{r} - \vec{r}_2)$

$\ddot{X} - \vec{r} - 2\vec{v} = \frac{-\mu}{r_1^3}$